GINsim user manual

Table of Contents

Introduction
- About GINsim
- Availability
- Running GINsim
- Overview of the manual
- New in GINsim 2.3

The User Interface
- The GINsim window
- File menu
- Edit menu
- View menu
- Actions menu
- The secondary window
- Processing a new graph

Editing the regulatory graph
- Notes
- Gene order
- Gene attributes
- Interactions
- Model integrity

The simulation
- Simulation Parameters
- Construction Strategy
- Initial states
- Mutant Definition
- Depth and size limitations
- Running the simulation

Analysing the regulatory graph
- Stable State Search
- Functionality of Regulatory Circuits
- Petri Net Export

Analysing the state transition graph
- Strongly connected component
- Path finder
- Animation plugin
- Trajectory visualization with gnuplot

References

Introduction

About GINsim

GINsim is a tool supporting the definition, the simulation and the analysis of regulatory graphs, based on a **multilevel logical formalism**. This formalism leans on the definition of two types of graphs: **regulatory graphs**, which model regulatory networks, and **state transition graphs**, which represent their dynamical behaviour, for given initial state(s) and under specific updating assumptions.

Availability

GINsim 2.3 is freely available, under the terms of the *General Public License*. The GINsim website (http://gin.univ-mrs.fr/GINsim) contains up to date informations, along with the latest official version of the software and its documentation.

Running GINsim

GINsim is a Java application and thus should run on any system offering a Java virtual machine (version 1.4 or later). If Java is not already installed on your computer, it can be downloaded:

- from sun Java page for Windows, Solaris and Linux (i386 and AMD64),
- from apple Java page for Mac OS X users,
- from IBM Java page for some other systems (AIX, linux-powerpc).

Other system requirements depend heavily on model size. To work confortably, a computer with at least 512Mo of RAM is required. A screen resolution of at least 1024x768 is recommended.

GINsim is available in two forms:

- **A single jar file**. On a properly configured system, double-click on the file to launch the application, otherwise run it with the command: *java -jar GINsim-$version.jar*.
- **An archive** (zip or tar.gz format). Uncompress it and run the included launcher script: runginsim.bat for windows, runginsim.sh for UNIX-like systems, and runginsim.command for Mac OS X. This version includes the documentation (tutorial and user manual) and can be extended through plugins.

Tip

Large models may require an increase of the amount of RAM available to the Java virtual machine. This can be done:

- for the jar file, with the command *java -Xmx300M -jar GINsim-$version.jar* (replace 300 by the amount of RAM you want, in Mo);
- for the archive, by changing the value of the *JAVAMAXMEM* variable in the launcher script.

Overview of the manual

The next section covers the Graphical User Interface of GINsim, mostly through the presentation of the main menu options. Section 3 presents the editing tools dedicated to regulatory graphs. Section 4 is dedicated to simulation (i.e. the generation of state transition graphs). Finally section 5 and 6 present analysis tools.

For a first introduction to GINsim, the GINsim tutorial may be a better starting point. This manual uses as much as possible the same examples and figures.

New in GINsim 2.3

Noteworthy improvements:

- "extended save" option when saving a file;
- redesign of the simulation dialog box;
- mutants definition replace transition blocking (and can be stored when using extended save).

New features:

- Petri net;
- analytic determination of the stable states of a regulatory graph;
- analysis of the functionality of regulatory circuits.

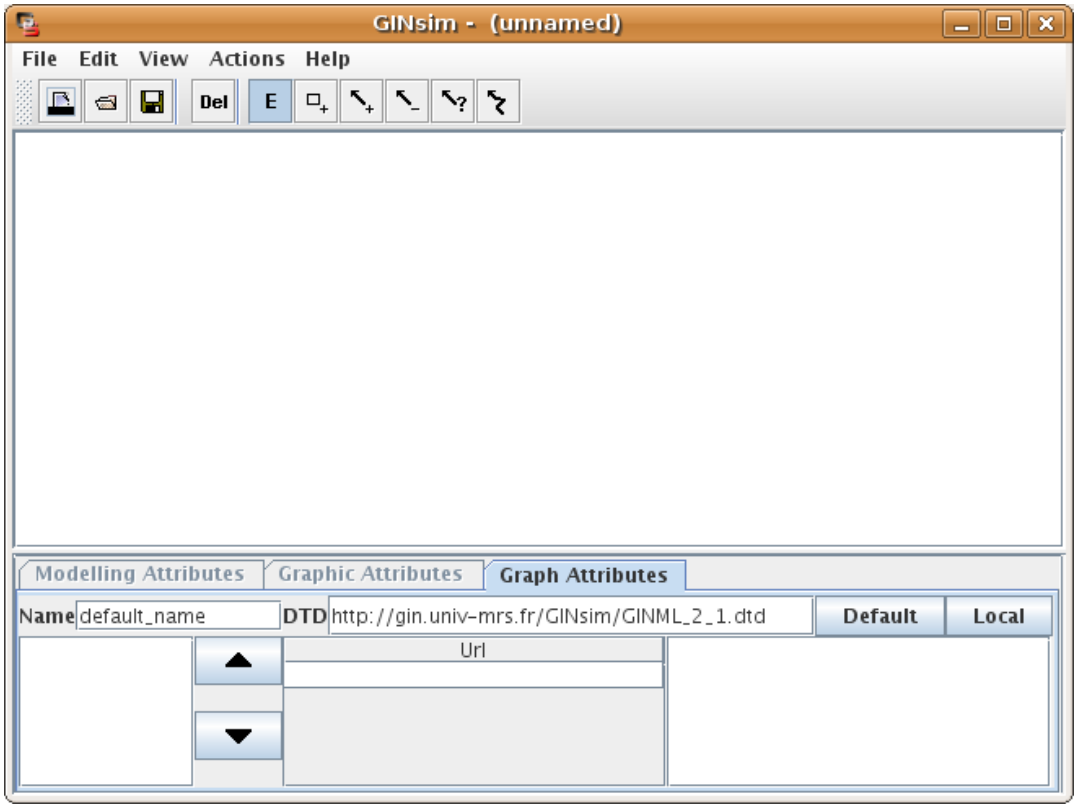
The User Interface

The GINsim window

When GINsim is launched, its main window opens (see Figure 1). This window is divided into three parts:

- the menu and toolbar on the top;
- the graph panel, as the main central part;
- the secondary panel on the bottom.

Figure 1. The main window of GINsim

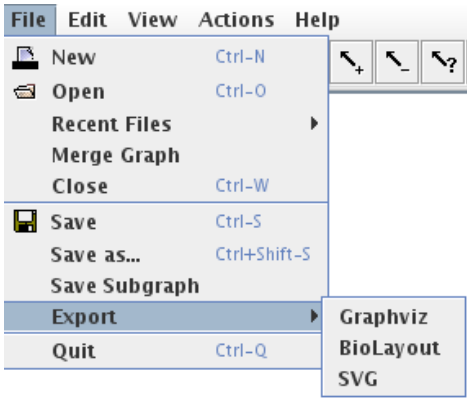


The main window of GINsim, featuring an empty model.

In the following, the contents of the menus are described.

File menu

Figure 2. File menu



The File menu offers all the classical options to open/save a file, to open/close a window, and to quit the application.

The File menu provides the following options:

- *New*: to create a new graph. This will open a new window unless the current graph is empty.
- *Open*: to load a graph from a file. This will open a new window unless the current graph is empty.
- *Recent Files*: to open a recently used graph. This submenu lists the more recently opened graphs.
- *Merge graph*: to open a graph and merge it with the current one. This option works only for regulatory graphs.
- *Close*: to close the current graph. If other windows are opened, it will simply close the current one, otherwise it will last with an empty window.
- *Save* and *Save as*: to save the current graph. If the file is new or if the *Save as* option has been selected, a file selection dialog appears (Figure 3) which allows to choose the graphical attributes to save: it is possible to save only the structure of the graph, ignoring all graphical attributes, or to save only the position of nodes. The default is to save all graphical attributes (position, size, color, shape...). The graph is saved in the (XML-based) **GINML** format.
- *Save Subgraph*: to save the current selection as a new graph.

- *Export*: to save the current graph, using another format. GINsim can export all kinds of graphs using several generic visualisation formats. These exports only retain the graph structure and visual appearance. The following export formats are available under the File/export submenu:

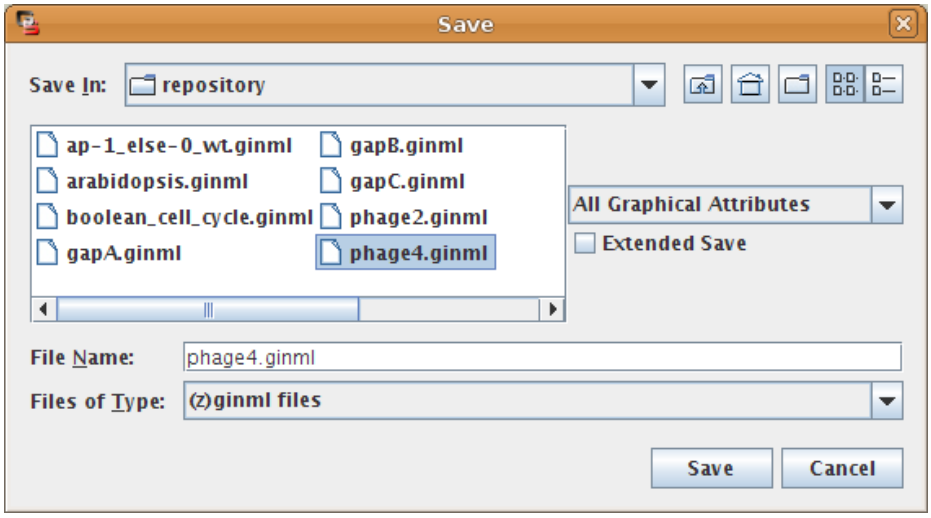
  - *Graphviz:* Graphs can be exported to the "dot" format used by the graph visualisation software Graphviz (http://www.graphviz.org). This export considers only the graph structure (graphical attributes are lost). It enables the display of large state transition graphs, that can not be viewed within GINsim.
  - *BioLayout:* graphs can also be exported to the BioLayout format (http://www.biolayout.org). Like in graphviz format, graphical attributes are lost.
  - *SVG:* the Scalable Vector Graphic (SVG) format is an XML-based format for representing vector graphics (http://www.w3.org/Graphics/SVG/). Tools like Inkscape (http://inkscape.org), can be used to display, modify and export SVG files to virtually any image format.

  The regulatory graph can additionally be exported to Petri net.

- *Quit*: close all graphs and exit the GINsim application.

Some of these actions (*New*, *Open* and *Save*) are also available from the toolbar.

Figure 3. The save dialog



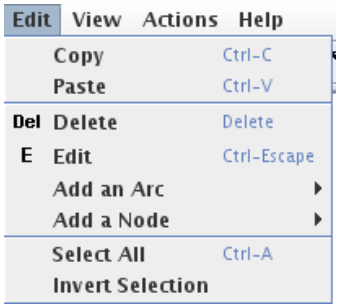
The Java *Save* dialog allows to browse and create folders, as well as to choose their location. By default, only folders and GINML files are shown; other files can be seen by removing the *GINML Files* filter. The choice list box on the right side allows to choose which graphical attributes to save. The *ExtendedSave* checkbox allows to enable or disable extended save (save as archive containing the graph and related data).

Tip

If the *Extended Save* option is selected, the file is saved in an archive (zip file with a .zginml extension) instead of a xml file (with a .ginml extension). This allows to save related data, such as simulation parameters or mutant definitions, along with the model. These files need GINsim 2.3 or later to be opened.

Edit menu

Figure 4. The Edit menu



The Edit menu allows the selection of some editing options. It is composed of three sections: copy/paste, graph editing tools and selection management.

Copy/paste

The edit menu offers classical *Copy* / *Paste* entries.

Note

Copy and *Paste* actions are specific to GINsim: copying the graph and pasting it into an external application is not supported. These actions are only available for regulatory graphs.

- Graph elements can be copied and pasted from one GINsim window to another.
- Pasted elements are automatically selected to ease their move.
- The *Copy* action does not test selected interactions, it will automatically select ALL interactions between selected genes.
- To avoid naming conflict, the identifiers of pasted genes will be changed by adding ".1", ".2"...
- Logical parameters are also copied and cleaned up: logical parameters involving not-copied nodes are suppressed. The resulting graph is consistent but these new parameters may need to be checked.

Graph editing tools

- *Delete*: to delete all selected elements.
- *Edit*: to select and move genes and arcs (default editing mode).
- *Add Node*: to add a node. This submenu lists the node insertion modes. When one of these modes is selected, clicking in the graph panel creates a new node.
- *Add Arc*: to add an arc. This submenu lists the arc insertion modes. When one of these modes is selected, dragging from a node to another adds an arc between them.

Toolbar buttons show the selected editing mode and allow to change it.

Tip

The selection of non-default editing modes is transient by default: once the action has been performed, GINsim switches back to the default mode (E). GINsim can be locked in any mode by double-clicking on the corresponding toolbar button.

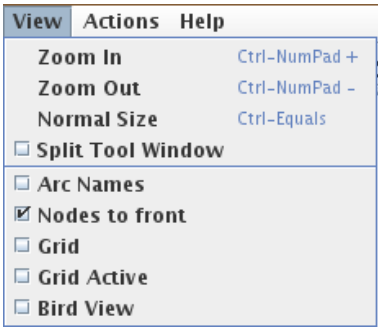
These options are **not** available when visualizing a state transition graph.

Multiple Selection

- *Select All*: select all elements of the graph.
- *Invert Selection*: unselect all selected elements and select all others.

View menu

Figure 5. The View menu



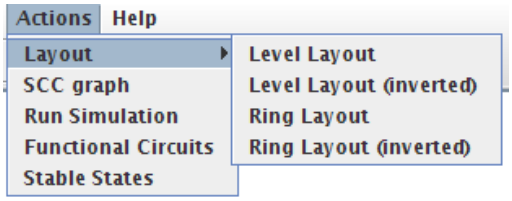
The View menu allows to control the zoom factor and to set some display options.

The View menu contains:

- *Zoom In/Out* and *Normal Size*: to control the zoom factor.
- *Split Tool Window*: to put the secondary panel in a separate window.
- *Arc Names*: to display or hide arc names (the source and target nodes).
- *Nodes to front*: to move all nodes to front (above arcs) or to back (makes selection easier).
- *Grid*: to display or hide the grid in the graph panel.
- *Grid Active*: to make the grid magnetic.
- *Bird View*: to display or hide the bird view (miniature) of the graph in the secondary window. The bird view gives an overview for large graphs, and enables the selection of the visible part.

Actions menu

Figure 6. The Actions menu



The Action menu for a regulatory graph.

Some actions can be performed from this menu, depending on the type of graph. Individual actions are detailed in the relevant part of this manual. Currently available actions are:

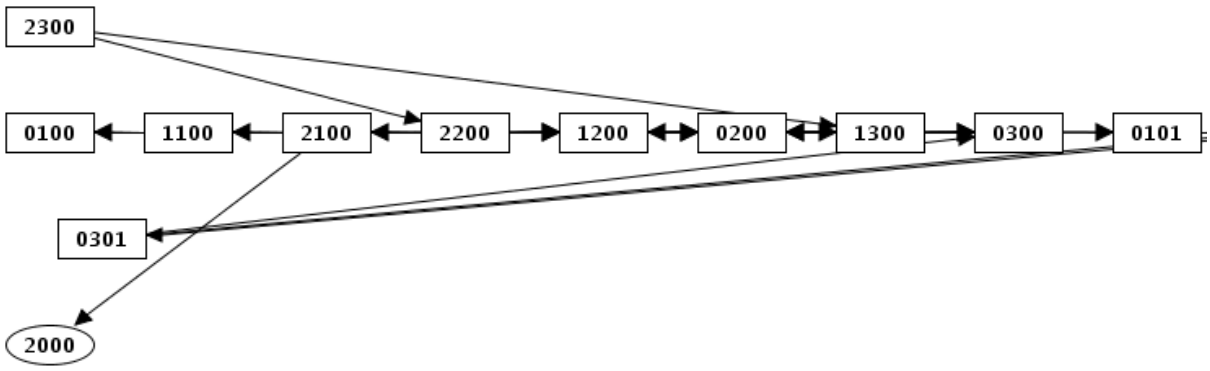
- for all graphs:
  - layout algorithms;
  - determination of the Strongly Connected Components (SCC) of a graph;
- for regulatory graphs:
  - simulation (*i.e.* determination of the state transition graph);
  - analyse of the functionality of regulatory circuits;
  - determination of stable states;
- for state transition graphs:
  - path search;
  - graphical path construction (animation).

Layout algorithms

Elements of the graph can be placed automatically (mainly useful for graphs constructed by GINsim, like state transition graphs).

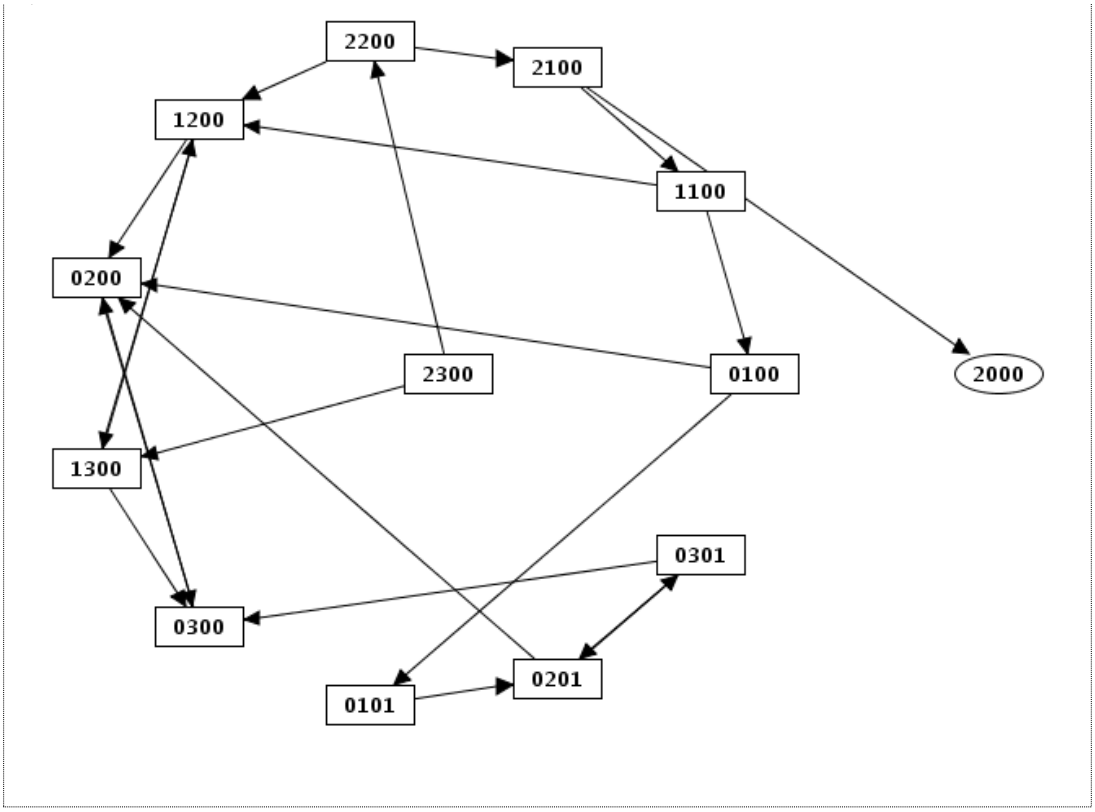
- Level layout: nodes are placed on rows, nodes without any incoming arc being placed on top and nodes without any outgoing arc at the bottom.

Figure 7. Level layout example



- Ring layout: nodes are placed on three concentric rings, source nodes at the center, terminal nodes at the periphery.

Figure 8. Ring layout example



Both the level and ring layouts have an inversed counterpart.

The secondary window

This window, which is originally attached to the main frame, offers a bird view of the graph and 3 tabs:

1. *Modelling Attributes* (this name changes according to the context),
2. *Graphical Attributes*,
3. *Graph Attributes*.

The *Modelling Attributes* and *Graph Attributes* tabs are specific to the type of graph and will be detailed later on.

Active tab change, depending on the current selection:

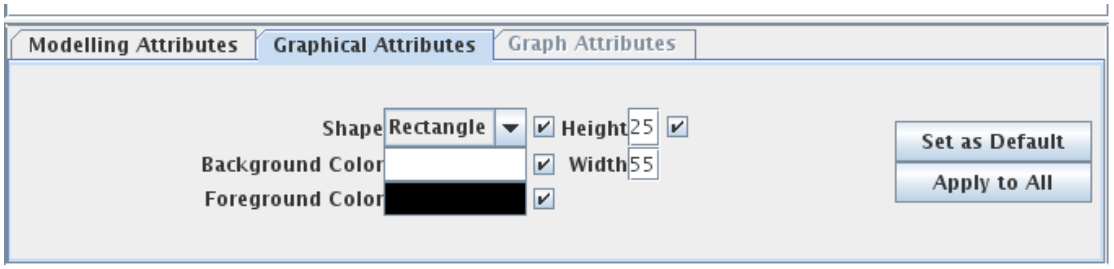
- if nothing is selected, only the *Graph Attributes* tab is active,
- if one node or one edge is selected, the first two tabs are active,
- if several nodes or several edges are selected, only the *Graphical Attributes* tab is active,
- if the selection contains both nodes and edges, all tabs are inactive.

Graphical Attributes

The *Graphical Attributes* tab in the secondary window allows to change the visual properties of the graph elements.

- Nodes:
    - background and foreground color,
    - shape and size.

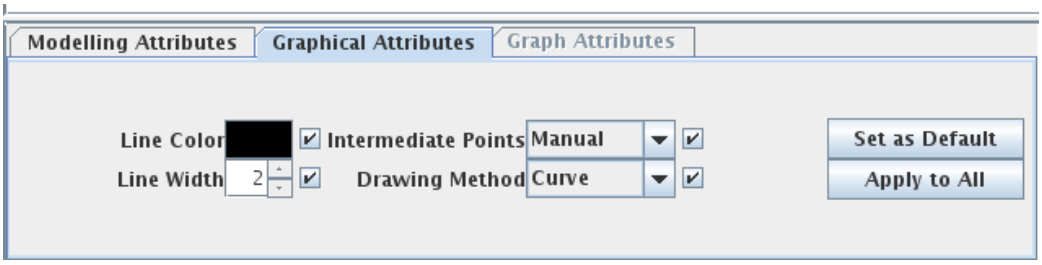
Figure 9. Graphical attributes for nodes



- Arcs:
  - line color,
  - line width,
  - intermediate points: they can be either automatically or manually placed,
  - drawing method: straight from one point to the next one, or following a bezier curve.

When the option for intermediate points is set to *Manual*, the points can be added by right-clicking on an arc. As the underlying drawing tool does not support *ctrl+click* to emulate right-click on Mac OSX, an additional toolbar button allows to insert them with a normal click ( ).

Figure 10. Graphical attributes for arcs



Tip

Buttons on the right side of the tab allow to apply the selected graphical attributes to additional elements:

- *Apply to All:* apply the selected attributes (those for which the checkbox is selected) to all nodes (or edges) of the graph. Note that, if several elements are selected, the selected attributes are only applied to the selection.
- *Set as Default:* use the selected attributes as default for elements that will be created in the future.

Bird view

The bird view offers an overview of the graph, a red rectangle highlighting the viewed part. It also offers some navigation facilities:

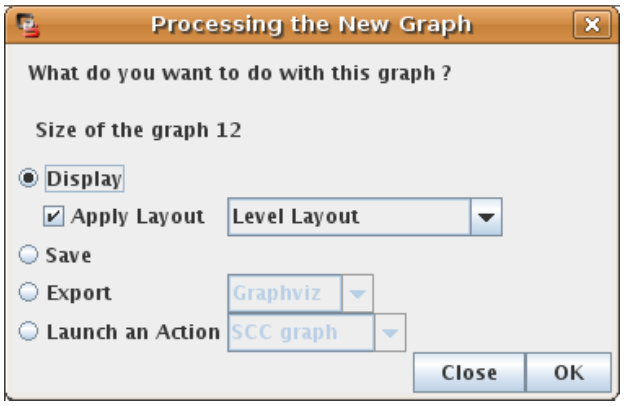
- click on it to change the viewed part,
- resize the red square (using the bottom right corner) to change the zoom factor.

Processing a new graph

When a graph is generated (state transition graph or graph of the strongly connected components), the user can choose to display it, to

perform some actions on it or to save it, through the *What to do* dialog box.

Figure 11. The *Processing the New Grah* dialog box



For state transition graphs, the number of stable states appears at the top-right corner.

Clicking on the "OK" button performs the selected action. This dialog box remains available to perform different actions on the graph. This dialog is closed when the graph is displayed, as all actions can then be performed through the *Actions* menu of the GINsim window.

Editing the regulatory graph

Regulatory graphs can be interactively modified: genes and interactions can be added, removed and their properties defined. Elements can be added, depending on the current editing mode:

Available editing modes for regulatory graphs	
Del	Deletion option: selected items are deleted.
E	Default editing mode: allows to select and move objects.
	Gene insertion mode: when selected, clicking on the graph panel adds a new gene.
	Interaction insertion mode: when selected, interactions are added by dragging from one gene to another. The interactions must be complemented by the definition of the logical parameters for the target variable (see below). The three buttons allow to add different kinds of interactions: activation, inhibition or undefined.
	Arc modification mode: when selected, intermediate points can be added or removed.

Note

The terms *gene* and *interaction* are used through this document, but nodes of the regulatory graph can represent a variety of *regulatory components* like protein concentration or qualitative state of the cell. Similarly, arcs can denote gene regulation but also protein phosphorylation, degradation, complex formation...

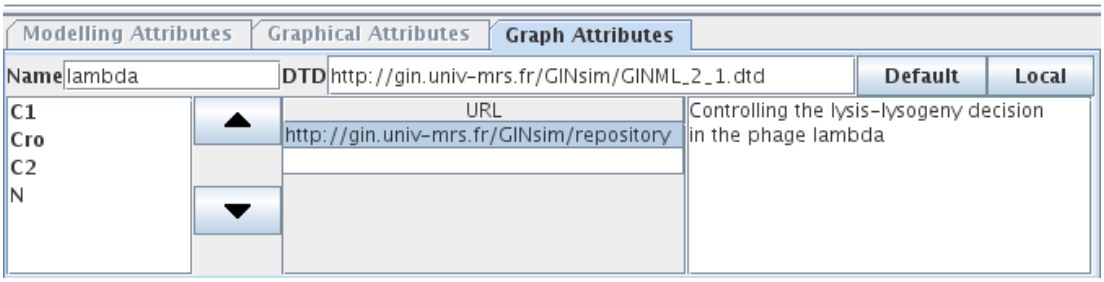
Notes

Notes can be attached to the different components of the regulatory graph:

- the graph itself,
- genes,
- interactions.

A note is composed of a textual comment and a list of URLs.

Figure 12. Notes

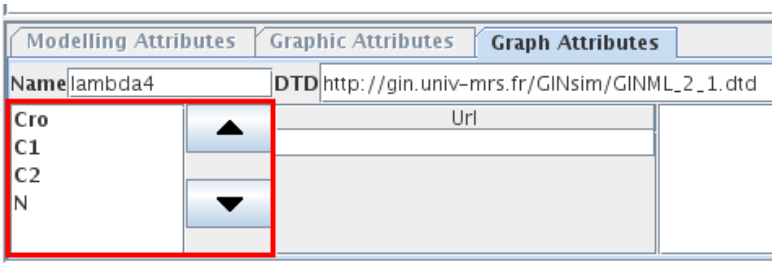


The same *Note panel* is used for all elements supporting notes. This screenshot shows this panel in the *Graph Attributes* tab.

Gene order

Genes are ordered in the regulatory graph. This order has no effect on the regulatory graph itself but may affect the (partial) simulation and the state transition graph (the same order being used in the states names). The gene order can be modified on the left side of the *Graph Attributes* tab.

Figure 13. Changing gene order



The left part of the *Graph Attributes* tab of a regulatory graph lists all genes of the model and allows to modify their order. The "up" and "down" buttons move selected genes in the list.

Tip

In all lists, several elements can be selected using the *ctrl* (*apple* on Mac OS X) or *shift* keys.

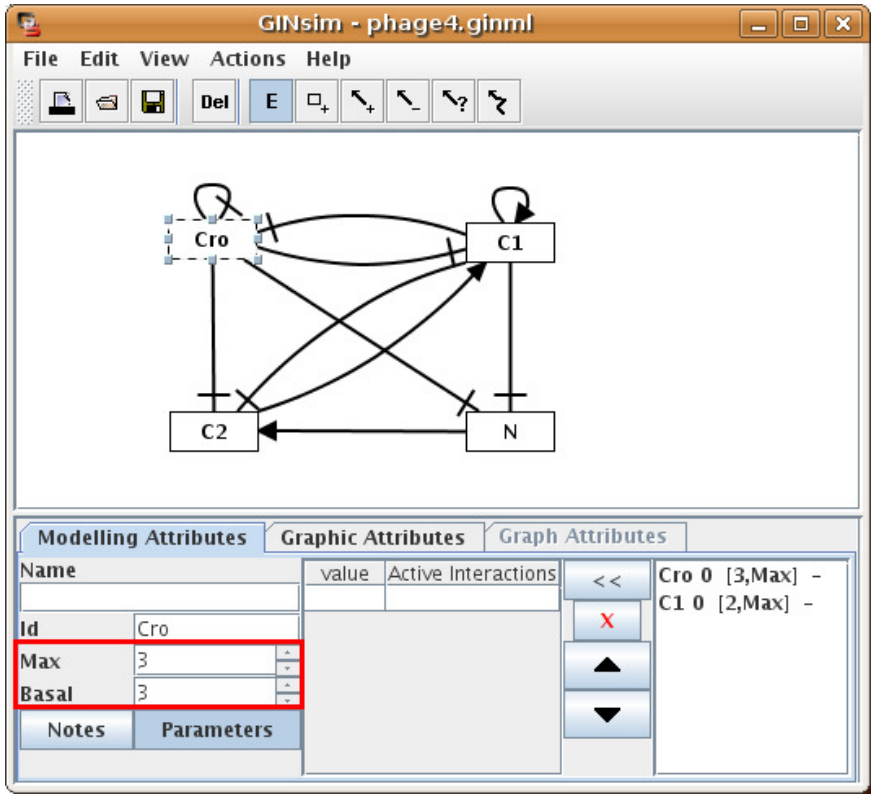
Several lists can be reordered using the "up" and "down" buttons.

Gene attributes

When a single gene is selected, the *Modelling Attributes* tab allows to define its properties:

- *Name*: long name of the gene, optional.
- *Id*: identifier of the gene, appears in the graph and must be unique.
- *Max value*: the maximal expression level of the gene. The default value is 1 (boolean case) and it can be increased for multi-valued genes.
- *Basal value*: the value towards which the node tends when none of its regulators is active. The default value is 0.
- *Logical parameters*: define the dynamical evolution of the expression level depending on the active incoming interactions.

Figure 14. Attributes of a gene



Properties of the gene Cro, as defined in the *lambda4* model.

Note

The *Modelling Attributes* tab is divided into two parts. The *Notes* and *Parameters* buttons (bottom left) respectively select notes and logical parameters editing.

Logical parameters

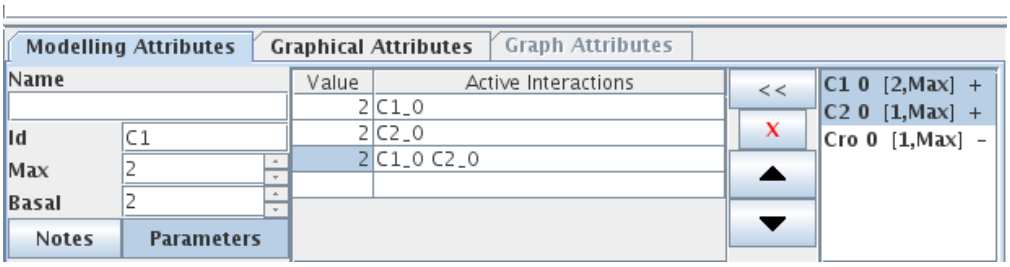
When a gene is selected, **logical parameters** defined for this gene can be configured in the right part of the *Modelling Attribute* tab. The panel dedicated to the definition of logical parameters is divided into three parts:

- On the left, a table lists all defined logical parameters, showing their values and related interactions.
- On the right, the list of all incoming interactions on the selected gene.

- The central part contains the following buttons:
  - "<<" assigns the set of selected incoming interactions (from the right list) to the selected logical parameter,
  - "X" deletes the selected logical parameter(s),

- "up" and "down" arrows reorder the selected logical parameters.

Figure 15. Definition of non zero parameters for C1



Show animation

The logical parameter panel, showing all parameters for C1.

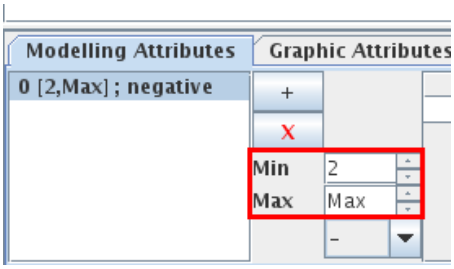
Tip

- While logical parameters can be numerous, usually many of them take the value zero. To shorten parameter definition, only non-zero parameters have to be defined in GINsim.
- Adding the same set of interactions twice is impeded.
- The set of interactions associated to a logical parameter can be modified: select the parameter and the corresponding set of interactions will be selected in the right list. This set can then be modified, and the "<<" button applies changes. If the new interaction set is already defined in the list of logical parameters, nothing is changed.
- To add a new logical parameter when an existing one is already selected, select the empty line at the bottom of the list.

Interactions

When a single arc is selected, the *Modelling Attributes* tab allows to define its properties

Figure 16. Properties of an arc



Properties of the Cro-N arc in the *lambda4* model.

As a gene can have different effects on another one, depending on its activity level, an arc can contain several interactions, listed on

the left.

- The "+" button adds an interaction.
- The "X" button deletes the selected interaction.

- Properties of the selected interaction can de defined:
    - *Min* and *Max* define its *range of activity*. The interaction is **active** when the activity level of its source gene is in this range.
    - *Sign*: is this interaction an activation, an inhibition or unknown ? This is only a visual hint, the real effect of interactions is defined through logical parameters.

Model integrity

GINsim keeps the definition of regulatory graphs consistent:

- When an interaction is deleted, all logical parameters in which it was involved are also deleted.
- When the max value of a node is decreased, interactions and logical parameters are checked and, if necessary, updated silently to avoid inconsistencies.

These automatic changes are necessary to keep the model valid, but they can alter it. Keep in mind that you need to double-check parameters after performing such changes.

Important

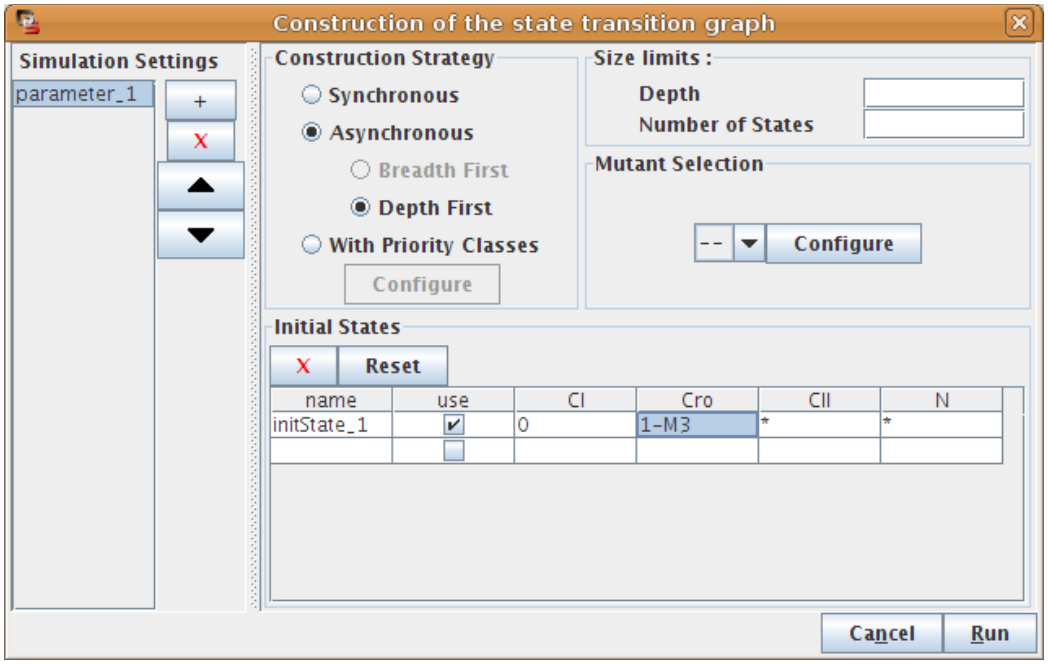
When a gene has several actions on another one, the activation ranges of these interactions should usually be disjoint, but GINsim does not check this property.

Checking activation intervals of interactions and the correctness of logical parameters is left to the user as adding more controls generates more annoyances than real help. Invalid logical parameters are highlighted to ease their detection. Keep in mind that a change in the activation-range of one of the interactions can turn a valid logical parameter into an impossible one. A parameter involving interactions from the same source with disjoint activity ranges would never be used and thus should not be defined.

The simulation

Once a regulatory model has been defined, a simulation can be launched through the *Run Simulation* option of the *Action* menu. This option triggers a dialog box allowing to choose simulation settings (Figure 17).

Figure 17. Simulation settings



This dialog box allows to configure and run the simulation.

Simulation Parameters

The left part of this dialog allows to manage a set of simulation settings: several simulation settings can be created, allowing to switch quickly from one to another. These settings will be saved and restored with the graph *if the extended save is selected*, otherwise they are lost when opening again.

The right part of the dialog box enables the definition of the selected setting.

Construction Strategy

In a given state of the system, one or several genes are called to change their values. When several changes are pending, different construction strategies lead to different successor states and thus to different state transition graphs.

Synchronous mode

In this mode, all changes are performed simultaneously. This simplification may generate artefacts in the state transition graph.

Each state has at most one successor state, which encompass all updating calls on gene activity levels.

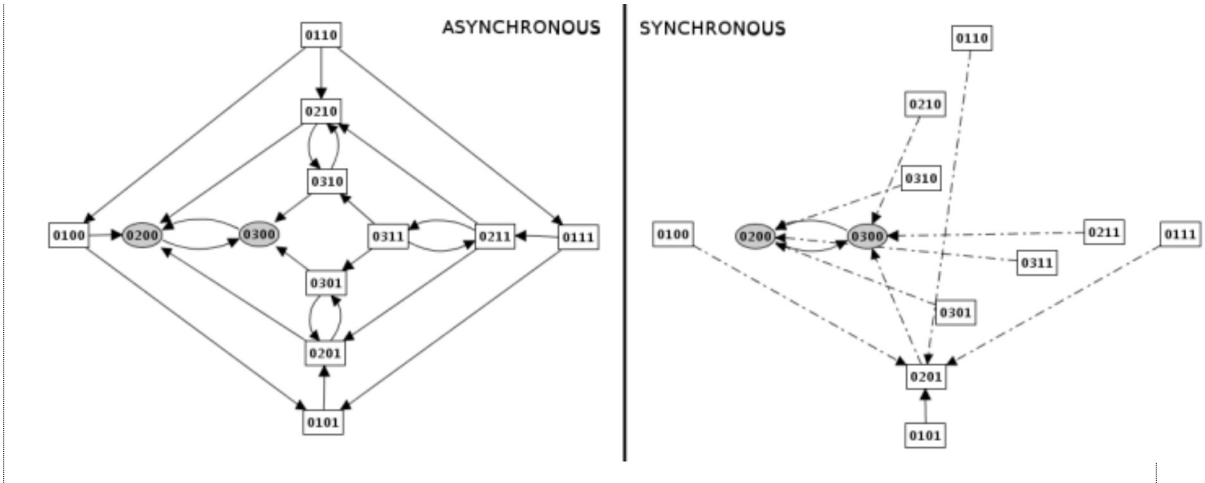
Asynchronous mode

In this mode, all changes are performed independantly. It will generate a state transition graph taking into account any possible trajectories. This mode is chosen by default.

Each state can have several successor states, each of them encompassing a single updating on one gene activity level.

In this mode, the graph transition state can be generated "depth first" or "breadth first". The same state transition graph will be built, except if interrupted (for illustration, see depth and size limitation).

Figure 18. Construction strategy: synchronous versus asynchronous



Samples of simulation results for the lambda4 model, applying synchronous and asynchronous strategies to the same initial states (all states where C1=0 and Cro>0). Dotted arcs denote that several transitions are triggered simultaneously.

Priority classes

This strategy allows the user to group components into different classes, and to assign a priority level to each of these classes. In case of concurrent transition calls, GINsim first updates the gene(s) belonging to the class with the highest ranking. For each class, the user can further specify the desired updating assumption, which then determines the treatment of concurrent transition calls inside that class. When several classes have the same ranking, concurrent transitions are treated under an asynchronous assumption (no priority).

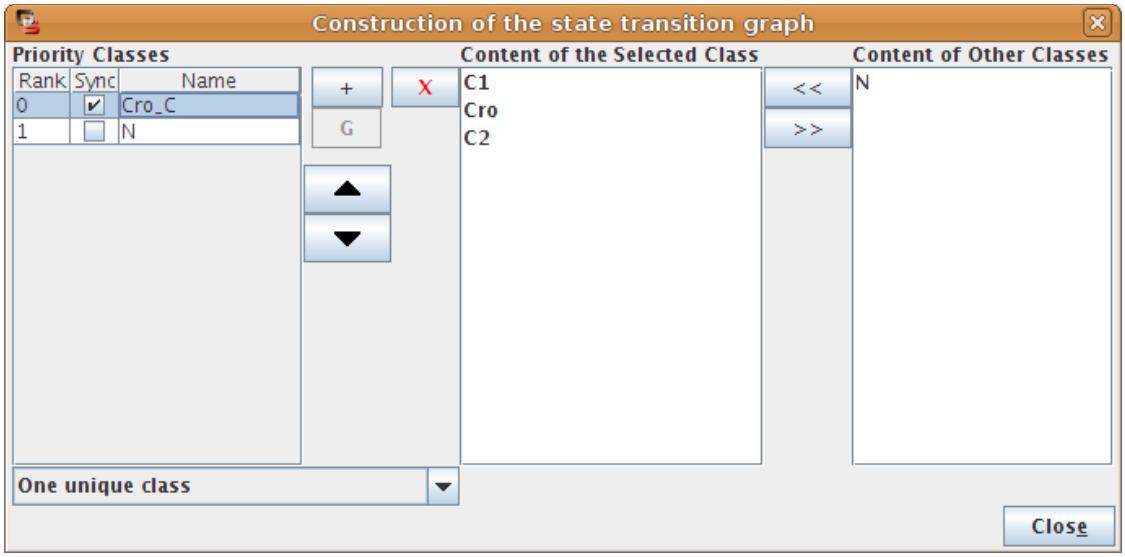
By default, all transitions belong to the same asynchronous class. Running a simulation without further configuration is thus the same as using the asynchronous assumption.

The left part of the configuration dialog box shows a list of priority classes (see Figure 19). The name of a class can be edited and a checkbox allows to change its internal mode from asynchronous (unchecked) to synchronous (checked). Buttons enable to add (+), delete (X), order (using the arrows) and group/ungroup (G) them.

The central column lists transitions that belong to the currently selected class and the column on the right lists all other transitions (belonging to other classes). To add transitions to the selected class, choose them in the right list an click on the "<<" button. The ">>" button removes the transition selected in the central list from the current class (and add them to the first class in the list).

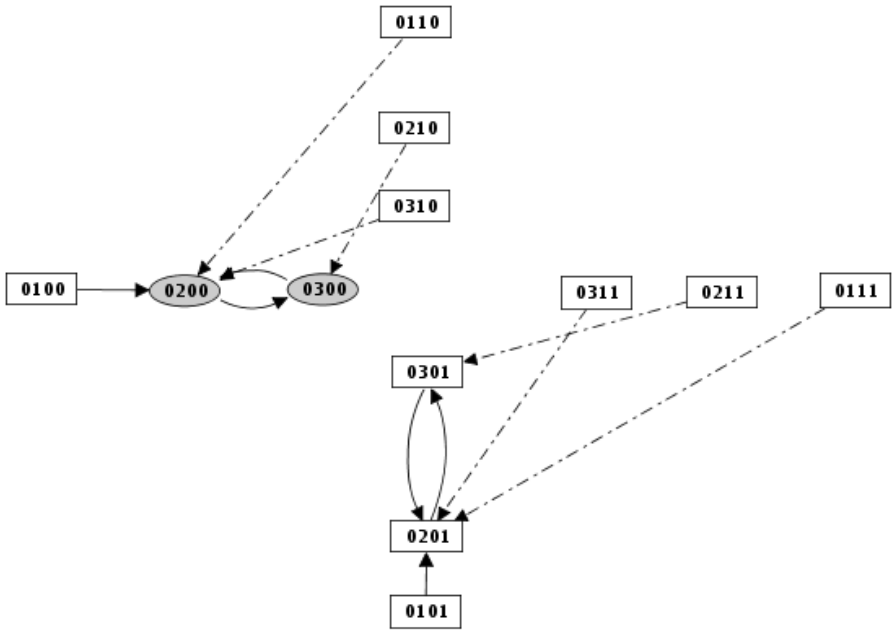
Finally, a choice list on the bottom left, allows to apply some predefined settings.

Figure 19. Definition of Priority classes



Show animation

Figure 20. Priority Class: example result



Example of simulation by priority classes. Two priority classes have been created, the highest ranked one is synchronous and contains C1, C2 and Cro. The other class contains only N. The resulting state transition graph is splited into two parts: N expressed vs. N not expressed.

Initial states

The bottom part of the dialog box is dedicated to the definition and the selection of the initial states of the simulation (see Figure 17). The main part is a table allowing to define initial states. Note that initial states are defined outside of the simulation setting and can thus be shared between several simulation settings.

Each row of the table corresponds to a set of states. This set is defined by the specification of allowed expression levels for each gene, in the corresponding table cell. Allowed expression levels are defined as semicolon (;) separated list of values or intervals (denoted by a dash). The special value "m" denotes the max value of the gene. examples: "0;2-4" means "0 or values between 2 and 4"; "0-m" means "any expression level". The default, denoted by a "*", is to not define any set ; it is thus the same as "0-m".

To be used for the current simulation setting, a line must be selected (using checkboxes in the second column). If no line is selected, all possible states are considered (resulting to the *full* state transition graph), otherwise only the states corresponding to the selected line(s) are considered.

The simulation will start with every selected initial states, unless they have already been reached by a previous simulation (i.e. from another initial states).

Note

The "X" button deletes the selected row of initial states. The "reset" button unselects all selected rows.

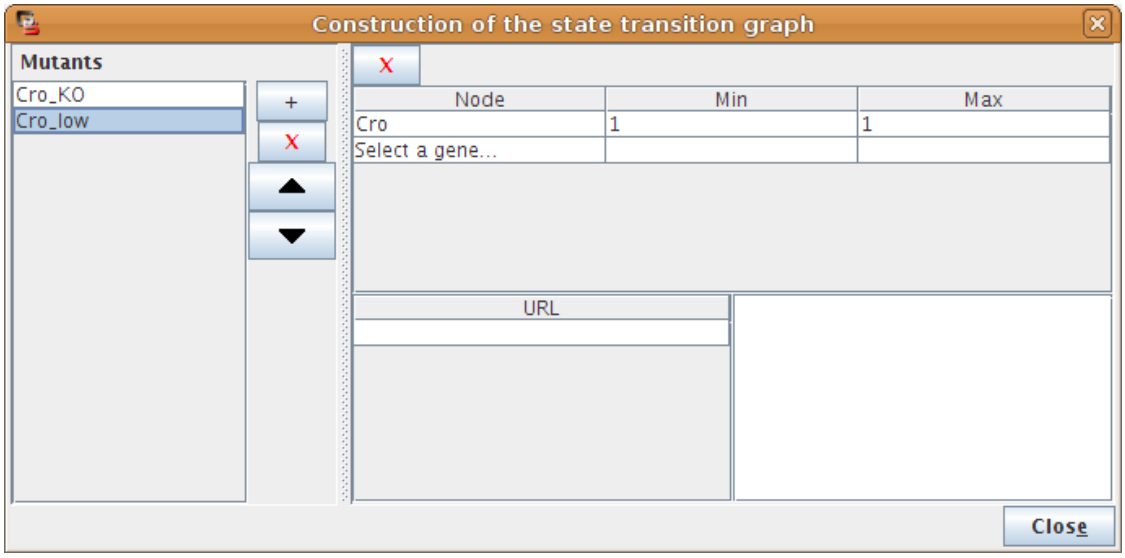
Mutant Definition

GINsim allows the definition of *mutants*, which can then be selected before running the simulation. A mutant is a set of restrictions on the evolution of the activity levels. The *Configure* button of the *Mutant* panel (in the simulation dialog box but also in some other places) opens a dialog box, allowing to specify minimal and maximal activity levels for some genes. Once the activity level of a gene enters the specified interval, it can not leave it anymore, i.e. transitions pushing the gene outside of the interval are ignored. This enables the definition of common mutants (knockouts or ectopic expressions for one or several genes) as an extension of the model. The simulation of more subtle mutants (conditional knockouts...) still requires modifications of the logical parameters.

Warning

These representation of mutants blocks some transitions, but does not affect trajectories where the activity level of the gene is outside of this range. Completely locking the expression level of a gene requires to set its initial values properly.

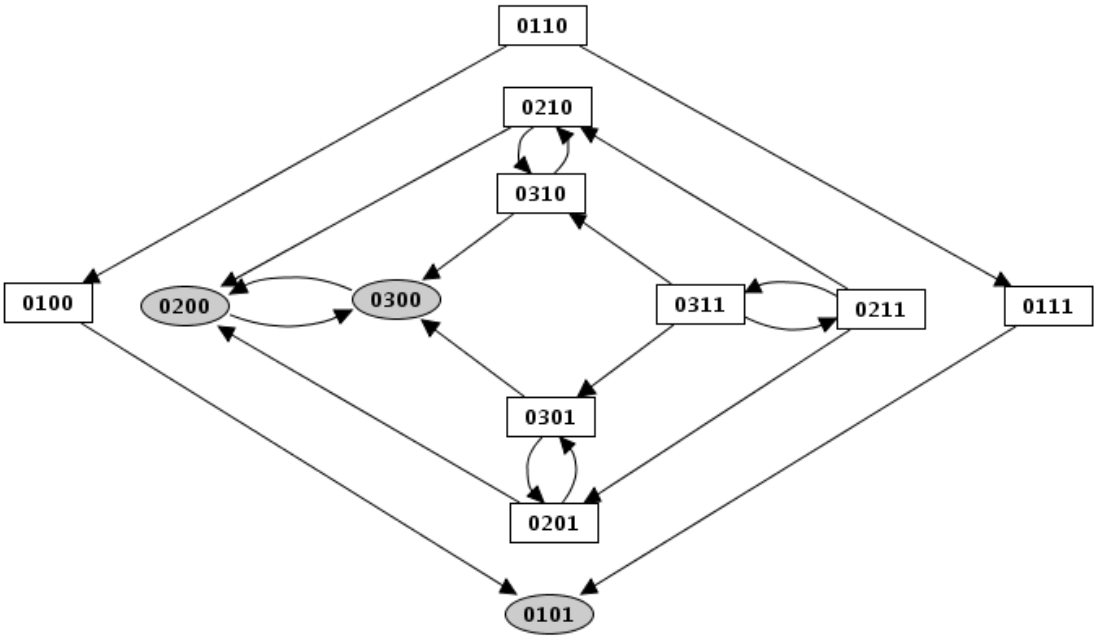
Figure 21. Mutant definition



Dialog box for the definition of mutants.

Show animation

Figure 22. Mutant simulation result



Result of an asynchronous simulation, where the expression level for Cro has been blocked at 1. The state transition graph is the same as the original asynchronous one, but all transitions where Cro leaves this value have been suppressed. This state transition graph is now composed of two disconnected parts, with a new stable state.

Depth and size limitations

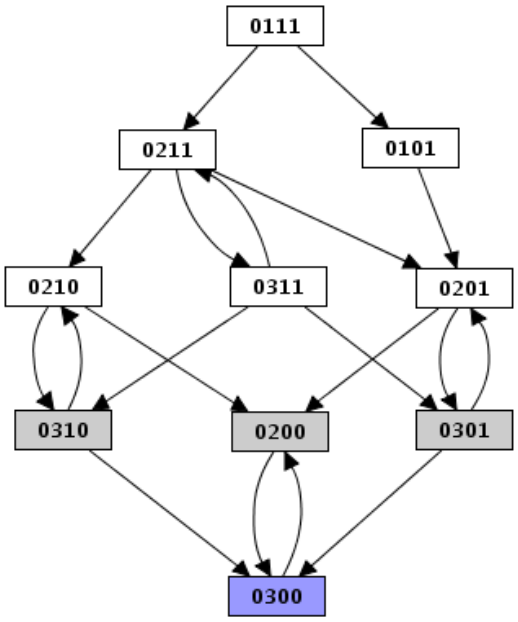
An option is also offered to limit the search depth and/or the total number of states generated.

The limit on depth applies to all simulation modes, excepting the asynchronous breadth first one (where no information on depth is stored).

Warning

When considering several initial states, some of them can be reached while running the simulation from an other state. In this case, no new search will be triggered for them and the depth counter will not be reinitialised (i.e. the depth limit for these initial states will be shorter).

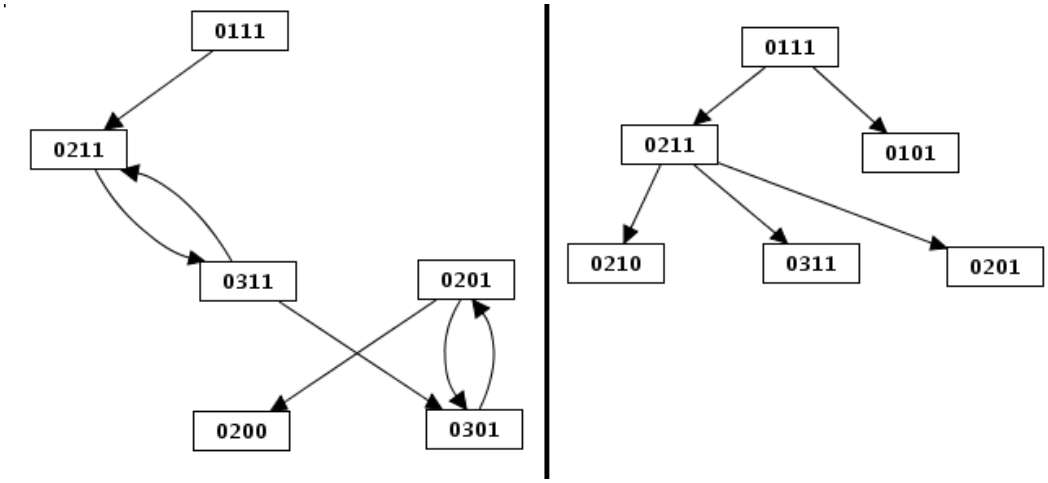
Figure 23. Limitation of the depth in the case of a depth first construction



State transition graph with all reachable states from the state "0111". The same simulation with a depth limit set to 2 keeps only the initial state and the nodes at a distance of two or less (i.e. the six white states).

The limit on the total number of states apply to all simulation modes. Under the asynchronous assumption, this limit has slightly different effects on depth first and breadth first search.

Figure 24. Limitation of the size (depth first and breadth first search construction)



The limit on the total number of nodes has different effects on depth first and breadth first state transition graphs. These examples show the graph of Figure 23 limited to 6 states. The first state transition graph was obtained using the depth first construction, whereas the second results from the breadth first one.

Running the simulation

While the simulation is running, the bottom left corner indicates the size of the generated state transition graph. The simulation can be interrupted, using the *Cancel* button, without loosing the calculated part of the state transition graph.

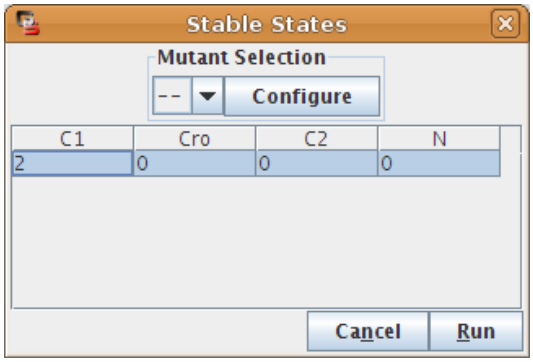
At the end of the simulation, the user can choose what he wants to do with the resulting state transition graph (see processing a new graph). The result are shown in a table the same dialog, allowing to launch the analysis again for another mutant. A "*" in the result table denotes that the state is stable for every value of the corresponding variable.

Analysing the regulatory graph

Stable State Search

The *Stable States* option of the *Actions* menu allows the analytic (i.e. without running a simulation) determination of stable states of the model. All stable states are determined, regardless of their reachability, using an algorithm described in [Naldi2007]. The stable states dialog box (see Figure 25) allows to run the analysis after the optional selection of a mutant. The result is shown in a table in the same dialog, allowing to launch it again for another mutant. A "*" in the table denotes that the state is stable for every value of the corresponding variable.

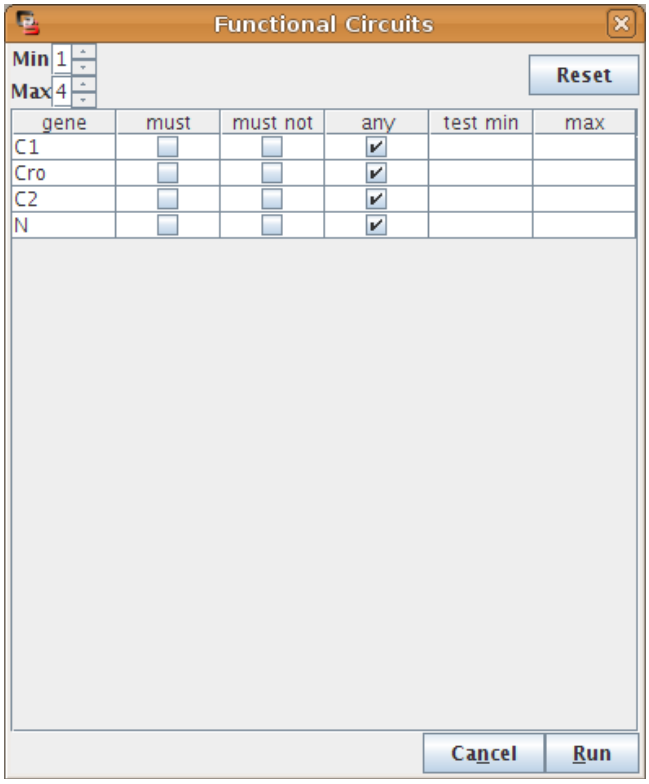
Figure 25. The stable states dialog box, showing the result of the analysis



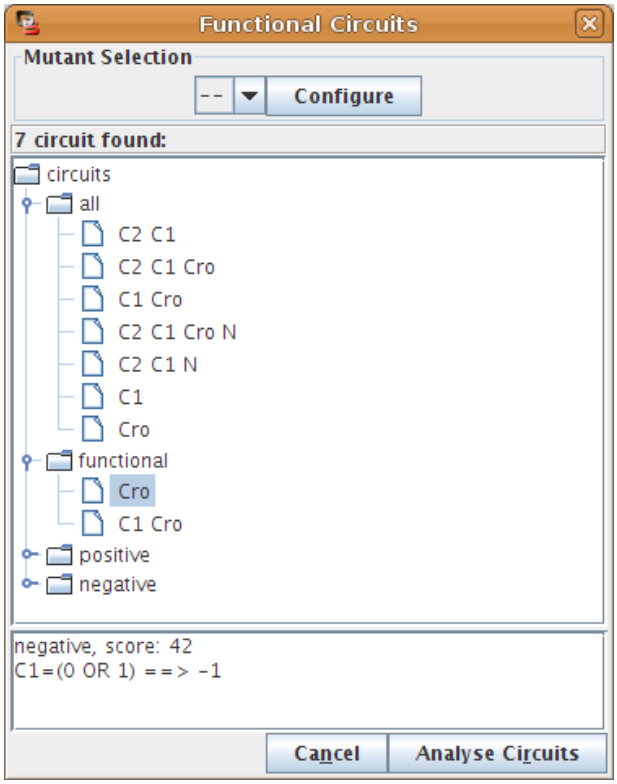
Functionality of Regulatory Circuits

Regulatory circuits have a crucial role in the dynamical behaviour of a system. In particular, positive circuits are required for the existence of several attractors and negative circuits for the presence of cyclic attractors. We presented in [Naldi2007] a method to analyse circuits of the regulatory graph. An implementation, described in Figure 26, of this analysis is available in GINsim in the *Circuits Functionality* option of the *Action* menu.

Figure 26. Finding and analysing regulatory circuits



The first dialog enables to select which circuits will be analysed, by specifying constraints on the length of the circuits or on the involved actors. By default all circuits are considered.

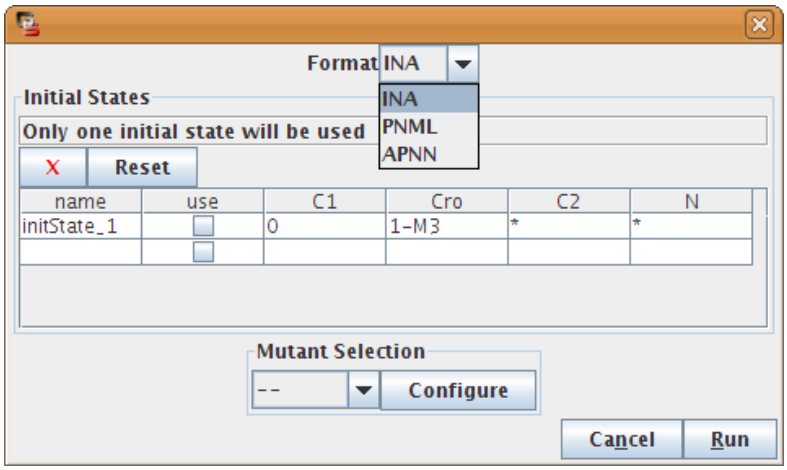


When this first step is completed, the second dialog shows the list of circuits and enables to run the analysis. When the analysis is completed, new categories appear to show which circuits are functional (or more specifically positive or negative). When a circuit is selected, its functionality context is shown in the bottom panel of the dialog. The analysis can be performed on the wild type or on any defined mutant.

Petri Net Export

Regulatory graphs can be exported into Petri net using the method described in [Chaouiya2006].

Figure 27. The Petri Net export dialog



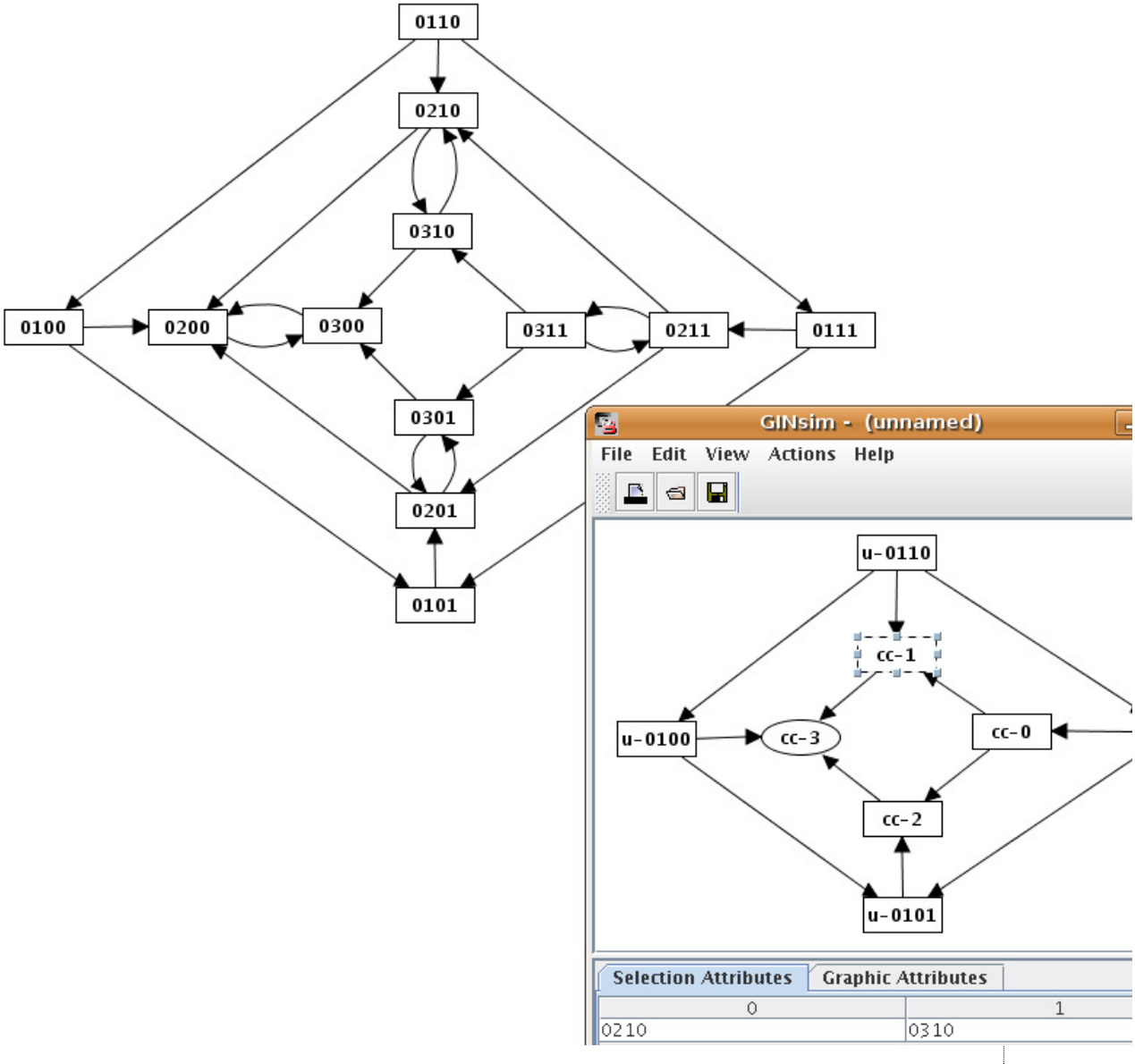
The Petri Net export dialog allows to select the export format, the initial markup and the mutant to apply before exporting. The initial markup can be taken from an existing initial state of the model (modifications made in this dialog are also applied to the simulation settings); only the first state is used.

Analysing the state transition graph

Strongly connected component

GINsim allows to compute the graph of the Strongly Connected Components (SCC graph, see http://en.wikipedia.org/wiki/Strongly_connected_component) of an existing graph, through the *SCC graph* option of the *Actions* menu. The SCC graph is an acyclic graph abstracted from the original one: each node of the SCC graph is a cycle or a set of intertwined cycles of the original graph. This makes it easier to follow trajectories as the resulting graph is more compact.

Figure 28. Strongly Connected Components graph



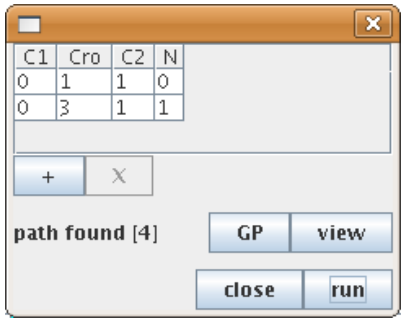
Example of Strongly Connected Component Graph. The original state transition graph is on top. The *Selection Attribute* tab in the bottom panel shows the content of the selected SCC (i.e. the list of nodes in the original graph).

Note that the *SCC Graph* tool can also be applied to regulatory graphs.

Path finder

The "search path" action allows to find a path between different states of the state transition graph. Selecting the "search a path" action in the action menu opens the configuration dialog. This dialog allows to specify the starting state, the target state , and intermediate states (using the "+" and "X" buttons to add/remove them). GINsim then performs a shortest path search (If several paths exist between these states, the result is thus the first encountered shortest path).

Figure 29. Search for a path in the state transition graph

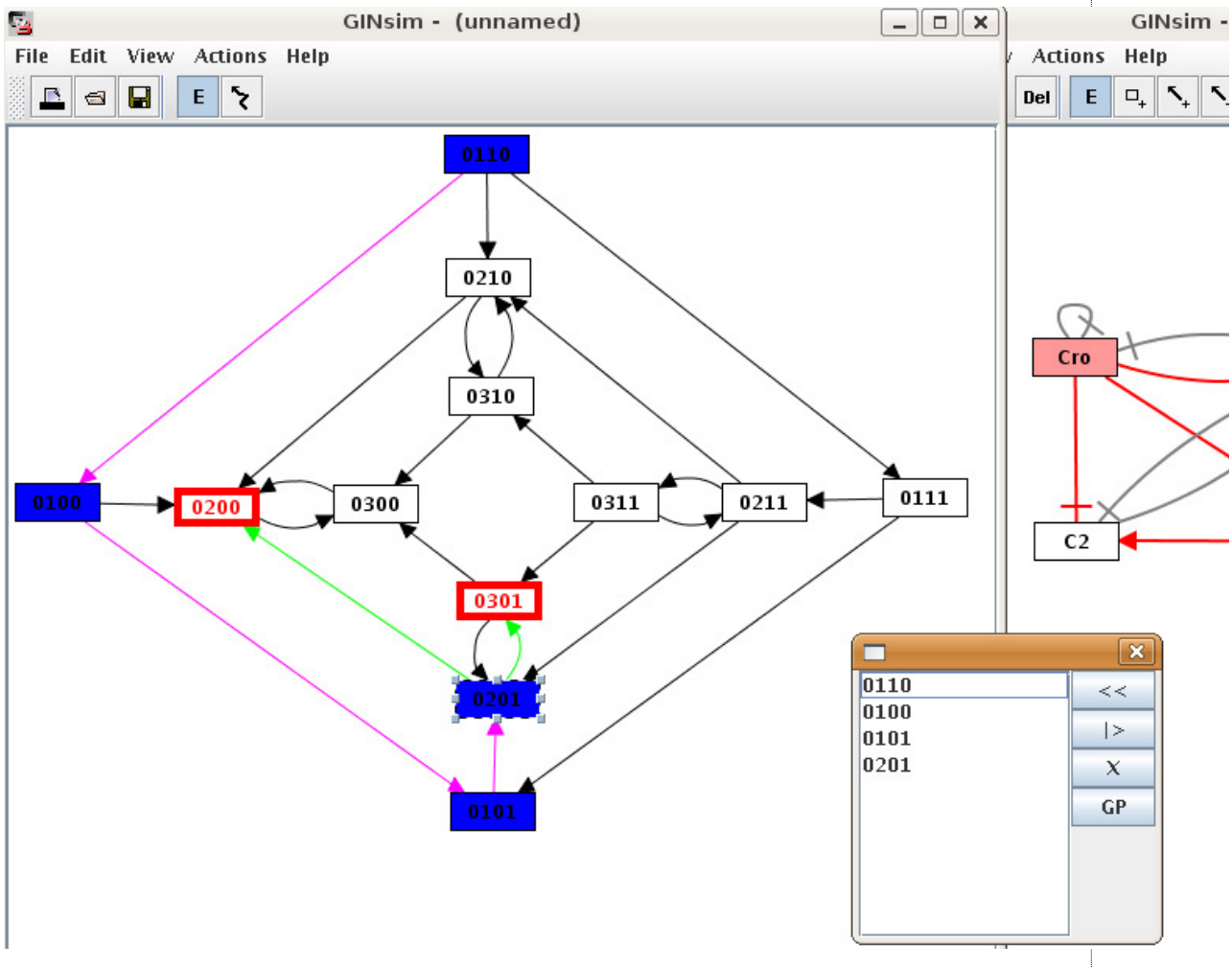


The search tool has found a path from the state *0110* to *0311*, involving four steps.

Animation plugin

The animation tool is launched through the *Animate* option of the *Action* menu of a state transition graph. It allows to interactively follow a trajectory in the state transition graph. Genes of the regulatory graph are highlited, according to the selected state.

Figure 30. The animation plugin



Show animation

To follow a trajectory in the state transition graph, select a starting state. It is then colorized in blue and added to the trajectory (listed in the animation dialog box). Its successor states are highlited, and can be added to the trajectory upon selection (which changes the available successors).

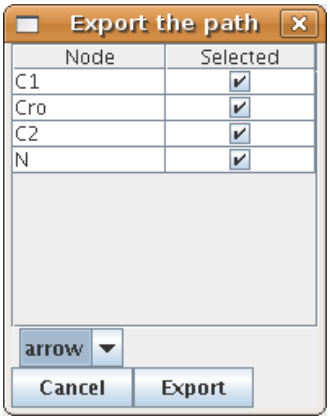
To build another trajectory, select a state in the dialog and press the "<<" button to remove it and all of its successors.

To view the defined trajectory, select the starting state and press the "|>" button.

Trajectory visualization with gnuplot

Path created with the animation plugin or found by the path finder can be visualized using gnuplot. The "GP" button (in the path finder and in the animator box) opens the gnuplot export box.

Figure 31. Gnuplot export box



The gnuplot export box allows to select genes that should appear on the graph.

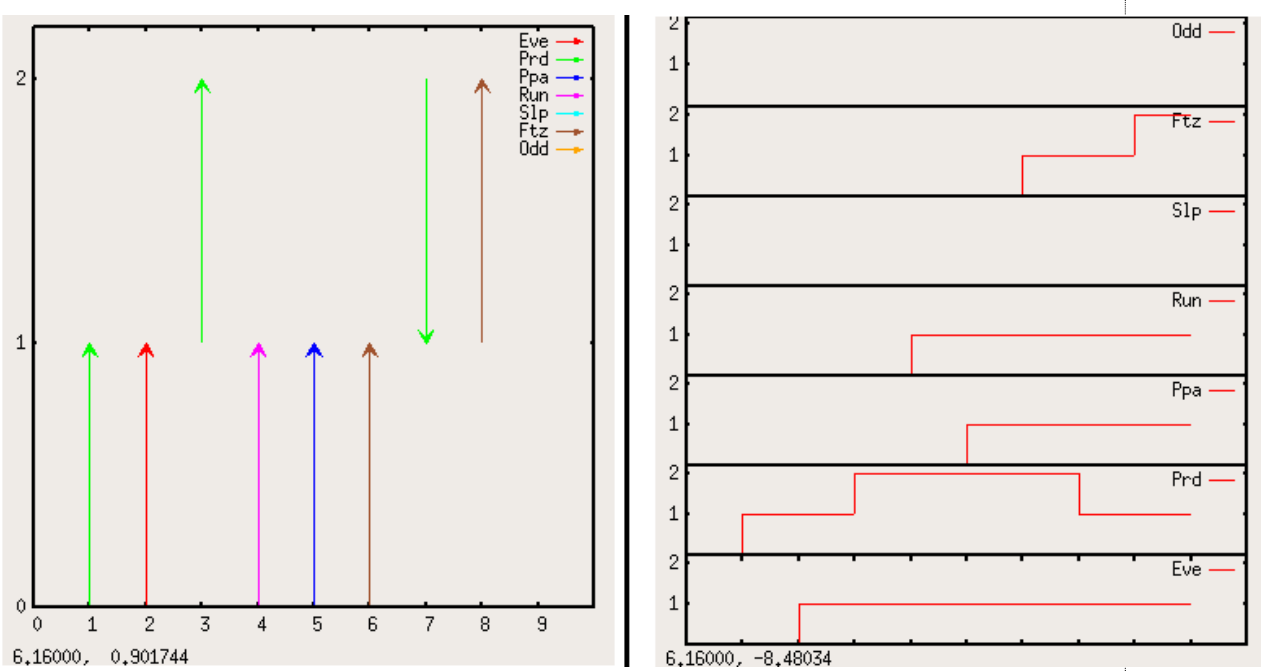
First select nodes that should appear in the graph, selecting many nodes badly affects the graphs.

Then select the kind of graph to create (see bellow).

Finally, click "export" and choose a place to save your plot. Note that two files will be created but only the first file name will be asked to the user.

If you saved it as "myplot.gnuplot" (the second file will then be named "myplot.data"), the graph can be obtained by running gnuplot and typing the *load "mygraph.gnuplot"* command at the gnuplot prompt.

Figure 32. gnuplot export examples



Examples of the two kinds of gnuplot graphs available. The first one uses arrows of different colors to show changes in the expression levels. The second one show separate subgraphs for each gene.

References

[Chaouiya2003] Claudine Chaouiya, Elisabeth Remy, Brigitte Mossé, and Denis Thieffry. *Qualitative analysis of regulatory graphs: a computational tool based on a discrete formal framework*. Lectures Notes in Control and Information Sciences. 2003. 294. 119-126.

[Chaouiya2006] Claudine Chaouiya, Elisabeth Remy, and Denis Thieffry. *Qualitative Petri Net Modelling of Genetic Networks*. Lecture Notes in Computer Science. 2006. 4220. 95-112. 10.1007/11880646_5.

[Gonzalez2006] Aitor Gonzalez Gonzalez, Aurélien Naldi, Lucas Sanchez, Denis Thieffry, and Claudine Chaouiya. *GINsim: a software suite for the qualitative modelling, simulation and analysis of regulatory networks*. Biosystems. 2006. 84. 91-100. 10.1016/j.biosystems.2005.10.003.

[Naldi2007] Aurélien Naldi, Denis Thieffry, and Claudine Chaouiya. *Decision Diagrams for the Representation and Analysis of Logical Models of Genetic Networks*. LNCS:LNBI. 2007. 4695/2007. 233-247. 10.1007/978-3-540-75140-3_16.